

## REMARKS

### Status of the Claims

- Claims 1-16 are pending in the Application after entry of this amendment.
- Claims 1-16 are rejected by the Examiner.
- Claims 1, 10, and 11 are amended by Applicant.

### Claim Rejections Pursuant to 35 U.S.C. §102

Claims 1-16 stand rejected pursuant to 35 U.S.C. §102(b) as being anticipated by U.S. Patent No. 5,873,080 to Coden et al. (Coden). The Applicants respectfully traverse the rejection.

Coden teaches:

The present invention is a system and method that uses a single query with one or more media types to search a collection of multimedia documents in a database and produce a single combined result. The combined result can be made to conform to user requirements and can be generated by applying general logical operations to the results of the interim results of each of the search engines.

The invention has a combined query section which allows a user to input a single query with more than one media type. The combined query has a query data structure which is submitted to a query interface. The query interface stores the different parts of the query and then parses the query to separate the query according to type. A query object is built for each query type. The query interface translates each of the query objects by query type into queries which are understood by the application programming interface that is designed for a particular search engine. The query interface then distributes the queries to the appropriate search engines. The search of each query type is performed by the appropriate search engine(s) and each search engine returns results which are converted to a collection of result objects. (col. 3 lines 33-53).

Amended Claim 1 recites:

A method of distributing portions of a query over two or more execution engines, the method comprising:

receiving an input query into cascaded analysis engines;  
identifying with a first analysis engine, a portion of the input query that can be processed by a first execution engine;  
compiling the identified portion of the input query forming a first compiled portion;

- rewriting the input query to form a first rewritten query wherein the identified portion of the input query is removed from the input query and replaced by a first placeholder;
- passing the first rewritten query to a second analysis engine;
- identifying with the second analysis engine, a portion of the first rewritten query that can be processed by a second execution engine; and
- compiling the identified portion of the first rewritten query generating a second compiled portion wherein the input query is distributed over the first execution engine and the second execution engine.

Applicant finds support for this amendment in Figure 3 of the as-filed application. The embodiment of Figure 3, and corresponding paragraphs 0036-0048, describe in detail a example three stage cascade of analysis engines. In brief, the first analysis engine in the cascade of engines inputs the query and determines which portion of the query the first execution engine can process. The first analysis engine then sends out that first portion of the query to be compiled. The input query itself is then rewritten by the first analysis engine by removing the first portion of the query that will be executed by the first execution engine and replaced with a placeholder. The rewritten query is thus the original input query with the removed first portion replaced with a placeholder. The rewritten input query is then forwarded by the first analysis engine to the second analysis engine.

The second analysis engine inputs the rewritten query and determines what portion of the rewritten query the second analysis engine can process. The second analysis engine then sends that second portion of the query off to be compiled. The rewritten query that was received by the second analysis engine is then further rewritten by the second analysis engine by removing the second portion of the query and replacing the second portion of the query with a second placeholder. This twice-rewritten query is thus the received rewritten query with the second removed portion replaced with a second placeholder. The twice-rewritten query is then forwarded to the third analysis engine. The process continues to a third analysis engine. Thus it can be seen that the analysis engines are cascaded and that each analysis engine identifies a portion of its received query, sends that portion to be compiled, and then rewrites its received input query, and forwards the rewritten query to the next succeeding stage of the cascade of analysis engines.

In a compatible manner, each execution engine is then able to execute its received portion of the query and forewords its result to the next higher execution engine so that the last execution engine in the cascade of execution engines produces a full result.

Amended Claim 1 recites the cascade of analysis engines and the steps outlined above. Applicant notes that Coden does not teach a cascade of analysis engines. In contrast, Coden teaches that there is one component (the EUIS) that performs a parsing of the input query according to type and then it distributes the typed query portions to the appropriate search engine.

Specifically, Coden teaches in col. 5, lines 5-58:

It is the role of the Enhanced User Interface Support program (EUIS) (120) to accumulate the various user inputs until enough data is specified to formulate a valid query. This implies that the EUIS stores the state of the user input. The EUIS (120) also parses the query into query objects each having a single media type and each being suitable for a particular search engine (162, 164, 166). These operations are well known.  
(col. 5, lines 5-58).

Thus, Coden teaches that one device, the EUIS, is responsible for parsing out multiple portions of a input query to multiple different search engines. This is in distinction to the present claims which recite receiving an input query into a cascade of analysis engines where each engine identifies a portion of the input query that can be processed by a specific execution engine, rewrites the input query, and passes the rewritten query onto the succeeding stage in the cascade.

Although the Examiner has cited Figure 6 and columns 11 and 12 of Coden to indicate a rewriting of the input query, this is an error. Coden in columns 11, lines 50 through column 12, lines 14 describes an example of how an operator, such as an AND or an OR, may be applied between the results if lists of result objects after the execution engines have collected results.

This is in contrast to Claim 1 which recites rewriting the input query to form a first rewritten query wherein the identified portion of the input query is removed from the input query and replaced by a first placeholder. Applicant notes that this rewriting step in Claim 1 is a rewriting of the input query, not the organizing of output results as in Coden in column 11-12.

Thus, Applicant respectfully submits that Coden does not teach rewriting the input query to form a first rewritten query wherein the identified portion of the input query is removed from the input query and replaced by a first placeholder and passing the first rewritten query to a second analysis engine in a cascaded system of analysis engines as recited in Claim 1.

Applicant has amended Claims 10, 11 to also include the aspect of a cascade of analysis engines where each engine rewrites the input query for the succeeding analysis engine.

Since Coden does not teach all of the elements of independent Claims 1, 10, and 11, then Coden cannot anticipate these claims or their dependent claims under 35 USC §102(b). Accordingly, Applicants respectfully request withdrawal and reconsideration of independent Claims 1, 10, and 11 and their respective dependent claims as they patentably define over the cited art.

**Claim Rejections Pursuant to 35 U.S.C. §103 (a)**

Claims 5 and 16 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,873,080 to Coden et al. (Coden) in view of US. Patent No. 6,834,287 to Folk-Williams et al. (Folk-Williams). Applicants respectfully traverse the rejection.

The teachings of Coden are discussed above. Folk-Williams teaches:

A classification engine provides flexible support for manipulation of attribute-based data by dynamic generation of SQL with classifiers constructed from different schema objects representing different database schemas. The classifiers may be constructed by defining classifiers corresponding to the database schema, and mapping the classifiers to columns on tables in the database. The invention also allows a classification system to modify the database structure and easily conform the classification engine to the modified structure without recompiling the engine or rewriting the application that use the classification system. The engine is conformed to the new structure by constructing a second schema object for the modified database. The schema objects are preferably defined using a field-based language such as extensible markup language (XML). (Abstract).

Folk-Williams fails to teach rewriting the input query to form a first rewritten query wherein the identified portion of the input query is removed from the input query and

replaced by a first placeholder and passing the first rewritten query to a second analysis engine in a cascaded system of analysis engines as recited in Claim 1.

Since both Coden and Folk-Williams fail to teach all elements of amended independent Claims 1 and 11, upon which Claims 5 and 16 depend respectively, then the combination of Coden and Folk-Williams cannot render obvious dependent Claims 5 and 16 according to MPEP 2143.03.

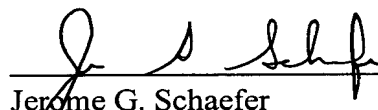
Applicant respectfully requests withdrawal of all rejections of the pending claims and requests reconsideration as the pending claims patentably define over the cited art.

**Conclusion**

Applicants respectfully submit that all pending claims patentably define over the cited art. Applicants respectfully request reconsideration and withdrawal of the rejections. A Notice of Allowance for all pending claims is requested.

Respectfully Submitted,

Date: September 5, 2006

  
Jerome G. Schaefer  
Registration No. 50,800

Woodcock Washburn LLP  
One Liberty Place - 46th Floor  
Philadelphia PA 19103  
Telephone: (215) 568-3100  
Facsimile: (215) 568-3439